

IN THE SPECIFICATION:

Please replace the paragraph from page 6, line 26 – page 7, line 18 with the following:

C |
--The stack mapper of the present invention seeks to determine the shape of the stack at a given PC. This is accomplished by locating all start points possible for a given method, that is, at all of the entry points for the method and all of the exception entry points, and trying to find a path from the beginning of the method to the PC in question. Once the path is found, a simulation is run of the stack through that path, which is used as the virtual stack for the purposes of the garbage collector. Accordingly, the present invention provides a method for mapping a valid stack up to a destination program counter through mapping a path of control flow on the stack from any start point in a selected method to the destination program counter and simulating stack actions for executing bytecodes along said path. In order to map a path of control flow on the stack, bytecode sequences are processed linearly until the control flow is interrupted. As each bytecode sequence is processed, unprocessed targets from any branches in the sequence are recorded for future processing. The processing is repeatedly repeated interactively, starting from the beginning of the method and then from each branch target until the destination program counter has been processed. Preferably a virtual stack is generated from the simulation, which is encoded and stored on either the stack or the heap. --

Please replace the paragraph from page 18, lines 22 - 32 with the following:

C2 -- Figure 5 illustrates a stack frame 500, having standard elements, such as an area for temps or arguments pushed by the method 502, ~~literals~~ literals or the pointer to the compiled method 504 (which also gives access to the stack map in the compiled method) and a back pointer 506 pointing to the previous stack frame. A small area of memory 508, possibly only four bytes, is left empty in the frame but tagged as needing dynamic mapping. An advantage of this is that if this stack frame 500 is deep in the stack, once the dynamic mapping has taken place, the frame will be undisturbed and is available for future activations. --